MODIFIED BAYESIAN CRAMÉR-RAO LOWER BOUND FOR NONLINEAR TRACKING

Onur Ozdemir§, Ruizin Niub†, Pramod K. Varshney†, and Andrew L. Drozdî

§ ANDRO Computational Solutions, LLC
7902 Turin Road Bldg 2 Rome, NY 13440, USA
† Department of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY 13244, USA

ABSTRACT

We propose a modified Bayesian Cramér-Rao lower bound (BCRLB) for nonlinear tracking applications where the prediction distribution conditioned on past measurements is used as the prior. The novelty of the proposed modified BCRLB comes from the fact that it utilizes past measurements, therefore it is specific to the current realization of the track which makes it a useful online tool that can be used for real-time sensor management. The computation of our proposed modified BCRLB is not analytically tractable except under very restricted conditions. Therefore, we also develop a particle based numerical computation method for our modified BCRLB so that this new bound can be easily calculated in real-time using the particles already available from the underlying particle filter which is used to track the target. We show by simulations that our developed numerical computation method approaches to its true analytical value as the number of particles in the particle filter increases.

Index Terms— Bayesian Cramér-Rao lower bound, nonlinear tracking, sensor management.

1. INTRODUCTION

The Bayesian (or posterior) Cramér-Rao lower bound (BCRLB) is defined to be the inverse of the Fisher information matrix (FIM) for a random vector [1] and provides a bound on the performance of estimators of the unknown target state for tracking applications. Only under very restricted conditions where the target motion model and the sensor measurement model are both linear and the noise for each model is Gaussian, we can get a closed form expression for PCRLB, which coincides with the error covariance of the Kalman filter. In [2], Tichavsky et al. provide a recursive approach for calculating the sequential PCRLB for a general multi-dimensional discrete-time nonlinear filtering problem. This lower bound gives an indication of performance limits, so it can be used as a criterion for off-line sensor management [3]. Many researchers have proposed PCRLB based approaches to solve sensor management problems. The PCRLB based approach essentially uses optimization techniques to control the measurement process, in order to minimize the lower bound on the root mean squared estimation error(s), determined from the FIM. In the traditional PCRLB, the FIM is derived by taking the expectation with respect to the joint distribution of the measurements and the target states. As a result, the useful measurement information is averaged out making the unconditional PCRLB an off-line bound which is independent of the current target track [4]. For this reason, we propose a new bound, which we call the modified BCRLB, that uses the prediction distribution from the tracking filter as the target state prior. Since our new bound uses the prediction distribution as its prior, it utilizes the information contained in the measurement data up to the current time for a particular realization of the state trajectory making it a track-adaptive bound. Since the proposed bound is track-adaptive, i.e., it is specific to a current track, it provides a more accurate and effective real-time performance evaluation than the traditional PCRLB. Therefore, this bound can be used to solve online sensor management problems such as sensor allocation, waveform design, search pattern management, quantizer design, bit allocation or a combination of these tasks.

Some attempts have been made in the literature to incorporate measurement information from the tracker to calculate different forms of PCRLB. A similar bound to our modified BCRLB has been derived in [4] called the conditional PCRLB. This bound is recursive, i.e., the bound at time \( k \) can be recursively calculated based on the bound at time \( k - 1 \), which is analogous to the traditional PCRLB. However, certain approximation has been made in [4], so that the conditional PCRLB becomes a recursive bound. In [4], it is shown by numerical examples that this recursive bound converges to the exact conditional PCRLB in a highly nonlinear filtering problem. However, in general, it is still possible that the approximation error is accumulated through propagation over time and results in divergence. In comparison, our modified BCRLB is not recursive in nature and is based on the target state distribution at time \( k \) only. It is also much simpler to implement than the conditional PCRLB.

In [5], a renewal strategy has been used to restart the recursive unconditional PCRLB, where the initial time is reset to a more recent past time. This allows the resulting PCRLB to be conditioned on the past measurements up to the reset initial time. It is difficult to evaluate the prior Fisher information, and one could use Gaussian approximation, which however may lead to large discrepancies. It is because a Gaussian approximation approach may not represent the actual posterior distribution for highly nonlinear scenarios where the resulting posterior distributions are multi-modal and/or have heavy tails.

The analytical calculation of our proposed bound is not tractable except for very restricted cases such as linear Gaussian systems. For this reason, we develop a numerical computation method to compute this new bound using the sequential Monte Carlo methods, i.e., particle filters [6]. With the recent advances in computation power, particle filters have become highly popular to solve nonlinear tracking problems. Our particle based computation method makes the modified BCRLB easily computable in real-time from the particles already available from the underlying particle filter which is used to track the target.

2. BAYESIAN SEQUENTIAL ESTIMATION AND PARTICLE FILTERING

Consider a \( n_x \) dimensional state vector at time \( k, x_k \), whose discrete time dynamics is defined by

\[
x_k = f_k(x_{k-1}, u_{k-1}),
\]

where \( f_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x} \) and \( u_k \) is the independent identically distributed (i.i.d.) process noise with dimension \( n_u \). The
measurement model is given by

\[ y_k = h_k(x_k, n_k) \]  \hspace{1cm} (2)

where \( h_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_n} \to \mathbb{R}^{n_y} \), \( n_k \) is the i.i.d. measurement noise, \( n_y \) and \( n_x \) are the dimensions of measurement and measurement noise vector, respectively. The process and the measurement noise distributions are denoted by \( p_{n_k-1}(n) \) and \( p_{n_k}(n) \), respectively. It is assumed that the estimator has complete information about the state dynamic model (1), the sensor measurement model (2) and the process and measurement noise distributions.

The Bayesian sequential estimation problem is defined as finding the belief in the state \( x_1 \) based on the measurements \( y_1 \) from time 1 to time \( k \), namely the posterior distribution (or the filtering distribution) \( p(x_k|y_1:k) \), where \( y_{1:k} = \{y_i\}_{i=1}^k \). Once the posterior distribution is calculated, the minimum mean squared error (MMSE) estimate of the state at time \( k, \hat{x}_k \), is given as its expectation with respect to this posterior,

\[ \hat{x}_k = \mathbb{E}_{p(x_k|y_1:k)}[x_k] \]  \hspace{1cm} (3)

The general Bayesian formulation consists of two stages which recursively calculate the posterior distribution of the state: 1) a prediction stage which propagates the posterior distribution based on the state dynamic model and 2) a filtering stage which uses new measurements to update the predicted posterior distribution. These stages can be expressed respectively as follows:

\[ p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1}) p(x_{k-1}|y_{1:k-1}) \, dx_{k-1} \]  \hspace{1cm} (4)

\[ p(x_k|y_{1:k-1}) = \frac{p(y_{k}|x_k) p(x_k|y_{1:k-1})}{p(y_{k}|y_{1:k-1})} \]  \hspace{1cm} (5)

where

\[ p(y_{k}|x_k) = \int p(y_{k}|x_{k-1}) p(x_{k-1}|y_{1:k-1}) \, dx_{k-1} \]  \hspace{1cm} (6)

It is known that Kalman filter provides an optimal solution to the Bayesian sequential problem for linear and Gaussian systems. In the case of some nonlinear/Gaussian systems, extended Kalman filter (EKF) can be used to provide a suboptimal solution by linearizing the nonlinear state dynamics and/or measurement equations locally, which can provide acceptable performance depending on the problem. For general nonlinear/non-Gaussian systems, particle filters have been shown to significantly outperform the Kalman filter and the EKF [7]. With the recent advances in computation power, particle filters have become popular tools for solving Bayesian sequential problems despite the high computational complexity associated with them. Particle filtering provides an approximate solution to the classical Bayesian sequential estimation problem by approximating the posterior distribution \( p(x_k|y_{1:k}) \) using a set of weighted samples, also called particles, \( \{x^{(j)}_k, w^{(j)}_k\}_{j=1}^M \), where \( w^{(j)}_k \) denotes the weight of the particle \( x^{(j)}_k \) at time \( k \) and \( M \) is the total number of particles [7],

\[ p(x_k|y_{1:k}) \approx \sum_{j=1}^M w^{(j)}_k \delta(x_k - x^{(j)}_k) \]  \hspace{1cm} (7)

### 3. POSTERIOR CRAMÉR-RAO LOWER BOUNDS

Let \( \hat{x}_k \) be an estimator of the state vector \( x_k \) at time \( k \), given all the available measurements \( y_{1:k} \). Then, the mean square error (MSE) matrix of the estimation error at time \( k \) is bounded below by the posterior (Bayesian) Cramér-Rao lower bound (PCRLB) \( J_k^{-1} \) [1, 2],

\[ E \left\{ [\hat{x}_k - x_k] [\hat{x}_k - x_k]^T \right\} \geq J_k^{-1} \]  \hspace{1cm} (8)

where \( J_k \) is the \( n_x \times n_x \) Fisher information matrix (FIM). In [2], Tichavský et al., provide a recursive approach to calculate the sequential FIM \( J_k \):

\[ J_{k+1} = D_k^{22} - D_k^{21} (J_k + D_k^{11})^{-1} D_k^{12} \]  \hspace{1cm} (9)

where

\[ D_k^{11} = E \left\{ -\Delta x_k \log p(x_{k+1}|x_k) \right\} \]  \hspace{1cm} (10)

\[ D_k^{12} = E \left\{ -\Delta x_k \log p(x_{k+1}|x_k) \right\} \]  \hspace{1cm} (11)

\[ D_k^{22} = E \left\{ -\Delta x_k \log p(x_{k+1}|x_k) \right\} + E \left\{ -\Delta x_k \log p(x_{k+1}|x_{k+1}) \right\} \]  \hspace{1cm} (12)

The operator \( \Delta \) in (10)-(12) is defined as \( \Delta = \nabla \log p(y_k | x_k) \), where \( \nabla \) is the gradient operator expressed as

\[ \nabla = \left[ \frac{\partial}{\partial x_1}, \ldots, \frac{\partial}{\partial x_n} \right]^T \]  \hspace{1cm} (13)

It is important to note that all the above expectations (10)-(12) are taken with respect to the joint probability density function (PDF) \( p(x_k) \).

\[ J_0 = E \left\{ -\Delta x_0 \log p(x_0) \right\} \]  \hspace{1cm} (14)

### 4. MODIFIED BAYESIAN CRLB

The traditional PCRLB can be calculated offline by using the target dynamic model in (1) and the sensor measurement model in (2). Then one can use the PCRLB to assess the bound on the estimation performance of the system and even design some system parameters off-line. However, from the dynamic point of view, it is not the best solution. If the information from a given tracker is intended to be used as a feedback for real-time sensor management, the useful measurement information for the current track needs to be utilized. In other words, one needs to take into account the information contained in the measurements up to time \( k \), i.e., \( y_{1:k} \) in order to find a bound on the predicted target state at time \( k+1 \). In essence, for a given track at time \( k \), all the information that is needed in order to design sensor management strategies for time \( k+1 \) is contained in the prediction distribution \( p(x_{k+1}|y_{1:k}) \), which is readily available from the tracking filter. If the tracking filter is a particle filter, \( p(x_{k+1}|y_{1:k}) \) is available in the form of propagated particles. Therefore, we propose a modified Bayesian CRLB [8], in which we use the prediction distribution at time \( k \), i.e., \( p(x_{k+1}|y_{1:k}) \), as our a priori distribution for time \( k+1 \). We denote our modified Bayesian CRLB as \( I_{k+1} \) where

\[ I_{k+1} = E_{p(x_{k+1}, y_{k+1}|y_{1:k})} \left\{ -\Delta x_{k+1} \log p(x_{k+1}, y_{k+1}|y_{1:k}) \right\} \]  \hspace{1cm} (15)

Using the fact that \( y_{1:k}, x_{k+1} \), and \( y_{k+1} \) form a first order Markov chain, the joint conditional density in (15) can be written as

\[ p(x_{k+1}, y_{k+1}|y_{1:k}) = p(y_{k+1}|x_{k+1}) p(x_{k+1}|y_{1:k}) \]  \hspace{1cm} (16)

Then \( I_{k+1} \) in (15) can be decomposed into two parts as

\[ I_{k+1} = I_{k+1}^{D} + I_{k+1}^{W} \]  \hspace{1cm} (17)

where \( I_{k+1}^{D} \) represents the information gained from the new measurements averaged over the a priori distribution, and \( I_{k+1}^{W} \) represents the information contained in the a priori distribution

\[ I_{k+1}^{D} = E_{p(y_{k+1}|x_{k+1})} p(x_{k+1}|y_{k+1}) \left\{ -\Delta x_{k+1} \log p(x_{k+1}|y_{k+1}) \right\} \]  \hspace{1cm} (18)
\[ I_{k+1}^P = E_p(x_{k+1} | y_{1:k}) \left\{- \Delta x_{k+1}^{t+1} \log p(x_{k+1} | y_{1:k}) \right\}. \tag{19} \]

Note that the expectations in (18)-(19) are taken with respect to the joint conditional density \( p(y_{k+1}, x_{k+1} | y_{1:k}) \). Therefore, \( I_{k+1}^P \) in (17) is in fact an expectation of the standard FIM over the prediction distribution \( p(x_{k+1} | y_{1:k}) \), where the standard FIM is defined by

\[ J_{k+1}^{\hat{p}}(x_{k+1}) = E_p(x_{k+1} | y_{1:k}) \left\{- \Delta x_{k+1}^{t+1} \log p(x_{k+1} | y_{1:k}) \right\}. \tag{20} \]

For a given track, our modified BCRLB can be calculated by adding \( I_{k+1}^P \) and \( I_{k+1}^l \) given in (18) and (19), respectively.

### 5. COMPUTATION OF THE MODIFIED BCRLB USING PARTICLE FILTERS

Notice that the distribution of \( x_{k+1} \) is represented by the propagated particles, \( x_{k+1}^{(j)} \sim p(x_{k+1} | y_{1:k}) \). In a standard particle filtering algorithm, the propagated particles are computed as

\[ x_{k+1}^{(j)} = f_{k+1}(x_k^{(j)}, \nu_k^{(j)}) \tag{21} \]

where \( \nu_k^{(j)} \sim p(\nu_k | \nu) \). Therefore, the prediction distribution is approximated using the weighted propagated particles \( \{ x_{k+1}^{(j)}, w_k^{(j)} \}_{j=1}^{M} \), i.e.,

\[ p(x_{k+1} | y_{1:k}) \approx \sum_{j=1}^{M} w_k^{(j)} \delta(x_{k+1} - x_{k+1}^{(j)}). \tag{22} \]

Using these propagated particles, we can approximate \( I_{k+1}^P \) as follows:

\[ I_{k+1}^P \approx \sum_{j=1}^{M} w_k^{(j)} J_{k+1}^{\hat{p}}(x_{k+1}^{(j)}), \tag{23} \]

where \( J_{k+1}^{\hat{p}} \) is the standard Fisher Information quantity defined in (20), which in many cases has a closed-form solution.

In order to compute the second term in (17), i.e., (19), \( p(x_{k+1} | y_{1:k}) \) should have a parametric expression. However, in our case, we have an approximation based on particles. In our previous work [8], we used a simple Gaussian approximation for the prediction distribution to approximate \( I_{k+1}^P \) such that \( p(x_{k+1} | y_{1:k}) \approx N(\mu_k, \Sigma_k) \) where

\[ \mu_k \approx \sum_{j=1}^{M} w_k^{(j)} x_{k+1}^{(j)}, \tag{24} \]

\[ \Sigma_k \approx \sum_{j=1}^{M} w_k^{(j)} (x_{k+1}^{(j)} - \mu_k)(x_{k+1}^{(j)} - \mu_k)^T. \tag{25} \]

Then, \( I_{k+1}^P \) is approximated by the inverse of the covariance matrix in (25)

\[ I_{k+1}^P \approx \Sigma_k^{-1}. \tag{26} \]

First rewriting the expression for \( I_{k+1}^P \) (19):

\[ I_{k+1}^P = E_p(x_{k+1} | y_{1:k}) \left\{- \Delta x_{k+1}^{t+1} \log p(x_{k+1} | y_{1:k}) \right\} \approx -\sum_{l=1}^{M} \Delta x_{k+1}^{t+1} \log p(x_{k+1}^{(l)} | y_{1:k}) w_k^{(l)}. \tag{27} \]

The second step of (27) is due to the approximation in (22). Notice that \( p(x_{k+1} | y_{1:k}) \) is given as:

\[ p(x_{k+1} | y_{1:k}) = \int p(x_{k+1} | x_k) p(x_k | y_{1:k}) dx_k \approx \sum_{j=1}^{M} p(x_{k+1} | x_k^{(j)}) w_k^{(j)}. \tag{28} \]

Using the approximation (28) in (27), the elements of the \( I_{k+1}^l \) can be computed as

\[ [I_{k+1}^l]_{mn} \approx -\sum_{l=1}^{M} \left( \frac{\partial^2 \log \left( \sum_{j=1}^{M} p(x_{k+1}^{(j)} | x_k^{(j)}) w_k^{(j)} \right)}{\partial x_{k+1}^{(l)} \partial x_{k+1}^{(m)}} \right) w_k^{(l)}. \tag{29} \]

Further simplifying (29) results in (30) and (31) given on top of next page. Note that in order for the approximation in (30) to be analytically tractable, the PDF that represents the state dynamics model \( p(x_{k+1} | x_k) \) should satisfy some regularity conditions:

1. \( p(x_{k+1} | x_k) \) exists and is finite for all \( k \) and \( i = 1, \ldots, n_x \).
2. \( p(x_{k+1} | x_k) \) is continuously differentiable.

As long as \( p(x_{k+1} | x_k) \) satisfies these regularity conditions, \( I_{k+1}^l \) (30) can be computed using the particles readily available through the particle filter which is used to track the target state. Note that the above regularity conditions are satisfied for most of the real tracking scenarios.

### 6. NUMERICAL RESULTS

In this section, we provide a numerical example to evaluate the accuracy of the approximation in (30) by simulations. We consider a simple linear Gaussian dynamic system, where we can calculate \( I_{k+1}^P \) analytically. The system model is given as

\[ x_k = x_{k-1} + \nu_{k-1}, \tag{32} \]

\[ y_k = x_k + n_k, \tag{33} \]

where \( \nu_k \sim N(0, 1), n_k \sim N(0, 1) \) and \( x_0 \sim N(5, 4). \) Since this is a linear Gaussian system, the optimal closed form Bayesian solution can be computed recursively using the Kalman filter. In this case, \( I_{k+1}^l \) can also be calculated in closed form and given as

\[ I_{k+1}^l = P_{k+1 | k}^{-1}. \tag{34} \]

where \( P_{k+1 | k} \) is the predicted (a priori) estimate covariance provided by the Kalman filter. In order to evaluate the accuracy of the approximation in (30), we implement a sampling importance resampling (SIR) particle filter [7] to track the target state. The number of particles is 2000. Figure 1 shows an example track, and the estimates provided by the Kalman filter and the particle filter. For this particular track, it is clear from Figure 1 that both estimates are very close to each other. We then use the same particles provided by the particle filter to approximate the \( I_{k+1}^l \) and compare it with \( P_{k+1 | k}^{-1} \) provided
\[ [I_{k+1}^m]_{mn} \approx - \sum_{l=1}^{M} \left\{ \frac{\partial^2 \phi(x_{k+1}^{(l)}, x_{k}^{(l)})}{\partial x_{k+1}^{l} \partial x_{k+1}^{n}} \phi(x_{k}^{(l)}) - \frac{\partial \phi(x_{k+1}^{(l)}, x_{k}^{(l)})}{\partial x_{k+1}^{l}} \frac{\partial \phi(x_{k+1}^{(l)}, x_{k}^{(l)})}{\partial x_{k+1}^{n}} \right\} w_{k}^{(l)}, \]

where

\[ \phi(x_{k+1}^{l}, x_{k}^{j}) = \sum_{j=1}^{M} p(x_{k+1}^{l} | x_{k}^{j}) w_{k}^{(j)}. \]

by the Kalman filter. Figure 2 shows the comparison plots for the corresponding track in Figure 1. The plots in Figure 2 show that the approximation is very close to the true value especially in the steady state. Table 1 provides relative approximation error values with respect to different number of particles. Relative approximation error is defined as the ratio of the absolute error values to the true value averaged over the track time. As can be seen from Table 1, relative error decreases as the number of particles increases. Although these results suggest that the approximation converges to the true value with increasing number of particles, we still need a formal analytical proof of this convergence which will be investigated in our future work.

**Table 1. Relative Approximation Error for \( I_{k}^m \).**

<table>
<thead>
<tr>
<th>Number of Particles</th>
<th>Relative Error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>5.8</td>
</tr>
<tr>
<td>250</td>
<td>3.9</td>
</tr>
<tr>
<td>1000</td>
<td>1.8</td>
</tr>
<tr>
<td>2000</td>
<td>1.4</td>
</tr>
</tbody>
</table>

7. REFERENCES


Acknowledgment

Ruixin Niu and Pramod Varshney’s work was supported by U.S. Air Force Office of Scientific Research (AFOSR) under Grant FA9550-10-1-0263.